HAL
open science

Bi-directional Recurrent End-to-End Neural Network Classifier for Spoken Arab Digit Recognition

Naima Zerari, Samir Abdelhamid, Hassen Bouzgou, Christian Raymond

To cite this version:

Naima Zerari, Samir Abdelhamid, Hassen Bouzgou, Christian Raymond. Bi-directional Recurrent End-to-End Neural Network Classifier for Spoken Arab Digit Recognition. ICNSLP 2018 - 2nd International Conference on Natural Language and Speech Processing, Apr 2018, Algier, Algeria. pp.1-6, 10.1109/ICNLSP.2018.8374374 . hal-01835440

HAL Id: hal-01835440

https://inria.hal.science/hal-01835440v1

Submitted on 11 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Bi-directional Recurrent End-to-End Neural Network Classifier for Spoken Arab Digit Recognition

Naima Zerari*, Samir Abdelhamid*, Hassen Bouzgou*, Christian Raymond†
*University of Batna 2, Algeria
{n.zerari,s.abdelhamid,h.bouzgou}@univ-batna2.dz
†INSA Rennes, IRISA/INRIA, France
christian.raymond@irisa.com

***Abstract*—Automatic Speech Recognition can be considered as a transcription of spoken utterances into text which can be used to monitor/command a specific system.**

In this paper, we propose a general end-to-end approach to sequence learning that uses Long Short-Term Memory (LSTM) to deal with the *non-uniform* sequence length of the speech utterances. The neural architecture can recognize the Arabic spoken digit spelling of an isolated Arabic word using a classification methodology, with the aim to enable natural human-machine interaction. The proposed system consists to, first, extract the relevant features from the input speech signal using Mel Frequency Cepstral Coefficients (MFCC) and then these features are processed by a deep neural network able to deal with the non uniformity of the sequences length. A recurrent LSTM or GRU architecture is used to encode sequences of MFCC features as a fixed size vector that will feed a multilayer perceptron network to perform the classification. The whole neural network classifier is trained in an end-to-end manner. The proposed system outperforms by a large gap the previous published results on the same database.

***Index Terms*—Arabic digits, Speech recognition, Auto-encoder, Mel Frequency Cepstral Coefficients, Long Short-Term Memory, Multilayer perceptron network.**

I. Introduction

Automatic speech recognition (ASR) is the process of understanding the human speech by a computer. In this context, Automatic Digit Recognition (ADR) is considered as one of the most challenging domains in ASR. The increasing importance of digit recognition is mainly due to the increasing demand for applications that deal with human-machine interaction through natural languages such as command systems via pronounced digit [1].

The implementation of these kinds of systems necessitate a specific process for the speech signal to deliver reliable features that can identify correctly the input speech utterances . Accordingly, a wide range of techniques have been proposed in the literature to parametrically represent the speech signal [2].

The most commonly used one, is the Mel-Frequency Cepstral Coefficients (MFCC), which is a popular technique that try to mimic some parts of the human speech production and speech perception [3].

In the present work, the resulted MFCC coefficients of the spoken Arabic digits will be fed to a Long Short-Term Memory (LSTM) architecture which deals with general sequence to sequence problems. The idea is to use the bidirectional LSTM layer to encode the sequence as a fixed size vector, then this vector will be introduced to a multilayer perceptron (MLP) classifier to perform the recognition task. The whole proposed model is trained in an end-to-end way with the aim to improve the recognition rate of the classification model on the arab digit dataset [4].

The remainder of the paper is organized in six sections as follows: Section II highlights some related works. Section III explains the methodology proposed in this study. Section IV presents the data used to validate the proposed methodology. Section V presents the performance criteria used to evaluate the proposed model. Section VI explains the experimental setup of the proposed classifier and presents the experimental results obtained on the data set, and compares with other existing approaches in the literature. Finally, section VII draws the conclusion of this work.

II. Related works

Several studies have been investigated in the literature to improve the recognition accuracy of an ASR system using different approaches [5].
Therefore, the number of research in Arabic language is limited compared to other languages such as English. In what follows, some studies concerning ASR systems for Arabic language will be discussed. The performance of each model varies in terms of accuracy.
In [6], the authors proposed a speech-and-speaker (SAS) identification system based on spoken Arabic digit recognition. They treated the speech signals as an image object and used conventional and Artificial Neural Networks (ANN) methods for classification and the algorithm of Teplitz matrix minimal eigenvalues as feature extraction method.
In [7], the authors described an automatic discrete speech

978-1-5386-4543-7/18/$31.00 ©2018 IEEE

Fig. 1: Block diagram of the proposed ASR system.

recognition system based on a tree distribution classifier. The MFCC feature extraction method was used to extract features followed by a vector quantization method (VQ). The VQ output was provided as an input to a classifier, which deliver the class-label according to each feature using an optimal spanning tree model in order to approximate the true class probability.

In [8], the authors introduced a fast learning method with a graphical probabilistic model for discrete speech recognition based on spoken Arabic digit recognition. The proposed method based on spanning tree structure takes advantage of the temporal nature of speech signal. The obtained results suggests that the proposed method was efficient in terms of time computation than the state-of-the-art algorithms that use the maximum weight spanning tree (MWST).

In [9], the authors proposed an Arabic digits classifier system with 450 Arabic spoken digits. The system is based on combining wavelet transform with the linear prediction coding method (LPC) to extract the features and the probabilistic neural network (PNN) for classification. The proposed classifier provided a high recognition rate, reaching about 93% of accuracy based on a speaker-independent system.

Recently, excellent performances on these systems have been achieved using Deep Neural Networks (DNNs) which are recent and extremely powerful machine learning models [10].

III. Methodology

Arabic is the native language of twenty-five countries including Algeria. It represents a Semitic language, and it is one of the oldest languages in the world. Currently it is the fifth language in terms of number of speakers [11]. The proposed system is based on the following steps, depicted in Fig.1 below.

A. MFCC feature extraction

To recognize the pronounced word; the speech waveform is converted into a parametric representation by using a specific technique. Thus, the reliability of Mel-Frequency Cepstral Coefficient (MFCC) technique to extract features from speech has been investigated. This technique introduced by *Davis* and *Mermelstein* in the 1980's, and have been widely used [12].

It is based on the frequency domain of Mel scale to capture the important characteristics included in the speech. Generally, MFCC is used to reduce the dimensionality of the acoustic signal while maintaining its discriminating power [13].

Thus, the feature extraction goal is to give a useful representation of the speech signal by capturing the significant information from it. The steps involved in MFCC feature extraction are: pre-emphasis, framing, windowing, Fast Fourier Transform (FFT), Mel scale filter bank analysis, logarithmic compression and discrete Cosine transform. The block diagram of the conventional MFCC is shown in figure 2.

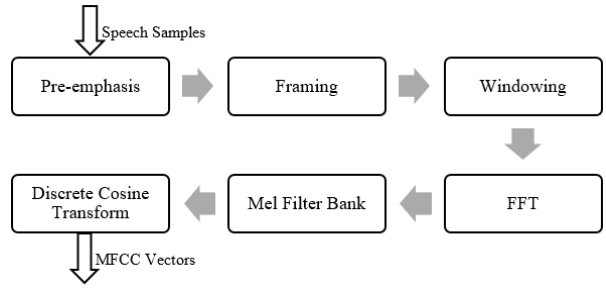


Fig. 2: MFCC Block diagram.

The main steps to compute MFCC coefficients are presented below:

- Pre-emphasized: The speech is, first, pre-emphasized with a pre-emphasis filter *1-az*$^{-1}$ to spectrally flatten the signal. In the time domain, the relationship between the output and the input of the pre-emphasis block is given in equation (1).

$$Y(n) = X(n) - aX(n-1), \tag{1}$$

where a is the factor which takes a value in the range $[0.9, 1]$, however, the default value is 0.97 [14], $X(n)$ is the input signal and $Y(n)$ is the output signal.

- Framing: the pre-emphasized speech is separated into short segments called frames. The frame length is sets to *20 ms* to guarantee stationarity inside the frame (Speech must remain stationary throughout the period of analysis). An overlap of 50% (10*ms*) is done between two adjacent frames to ensure the stationary between frames [14], [15].
- Windowing: is an important step to minimize the signal discontinuities at the beginning and the end of the each frame. For this propose, Hamming window is used to reduce the edge effect. It is expressed via the following mathematical expression [15].

$$Y(n) = X(n) * W(n), \tag{2}$$

$$W(n) = 0.54 - 0.46 \cos \frac{2\pi n}{N-1}, \tag{3}$$

where N represents the number of samples in each frame (window length) and n is from 1 to N.

- FFT: Fast Fourier Transform is applied to each frame to convert the N samples from time domain to frequency domain.
- Mel filter bank is applied to calculate the average energy in each block and take the logarithm of all filter bank energies by using one of the mathematical expressions below.

$$Mel(f) = 2595 * \log_{10}(1 + f/700), \quad (4)$$

or

$$Mel(f) = 1127 * \ln(1 + f/700), \quad (5)$$

Applying triangular filters on Mel-scale to the power spectrum to extract frequency bands. The Mel-scale aims to mimic the non-linear human ear perception of sound. These coefficients are highly correlated.

- The log Mel spectrum will be converted back to time domain. The result is called the Mel frequency cepstrum coefficients (MFCC). A good representation of the local spectral properties of the signal for the given frame analysis is delivered by the cepstral representation of the speech spectrum. Using the discrete cosine transform (DCT), the Mel spectrum coefficients are converted to the time domain. [16].

B. Recurrent neural networks and gated recurrent neural networks

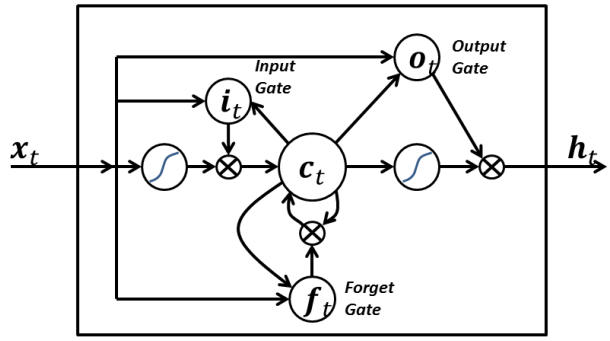
Recurrent Neural Networks (RNN) are a type of neural networks with recurrent connections. RNNs have been widely studied and used for problems involving input and output data of variable size and are particularly suitable for time series analysis and consequently used in automatic speech recognition [17], [18]. The success of this type of neural network is due to the specific variant, which are the long short-term memory (LSTM) proposed by Sepp Hochreiter and Jrgen Schmidhuber in 1997 [19] and the gated recurrent unit (GRU) proposed by Kyunghyun Cho et al in 2014 [17]. The main idea behind these networks is to use several gates to control the information flow from previous steps to the current steps [20].

By employing the gates, any recurrent unit can learn a mapping from one point to another. LSTM, contains three gates: an input gate, an output gate and a forget gate. At each iteration, the three gates try to remember when and how much the information in the memory cell should be updated [20]. A single LSTM memory cell is depicted in figure 3 [21] .

C. Multi-layer perceptron classifier

Multilayer perceptrons (MLP) is the most commonly used feedforward neural networks. The MLP architecture is variable, but generally organized in several layers of neurons. It consists of three sequential layers: input, hidden and output layer as depicted in figure 4. The input layer serves to feed the input vector to the network. The hidden layer output a non-linear transformation of the input in order to facilitate the determination of the last transformation computed by the output layer [22].

The MLP neural networks classifier acts usually in a supervised manner, to build an MLP classifier, a set of training data including the inputs and their associated outputs are requested. Hence, the classification is done by assigning a maximum value to the output neurons to represent the desired class [23].



Where

- $f_t = \sigma_g(W_f x_t + U_f h_t + b_f)$
- $i_t = \sigma_g(W_i x_t + U_i h_{t-1+b_i})$
- $o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o)$
- $c_t = f_t \circ c_{t-1} + i_t o \sigma_c(W_c x_t + U_c h_{t-1} + b_c)$
- $h_t = o_t \circ \sigma_h(c_t)$
- $\circ$ is Hadamard product

and:

- x_t = input vector, h_t = output vector
- c_t cell state vector
- W,U,b: parameter matrices and vector
- f_t, i_t and o_t
  - f_t: Forget gate vector: weight of remembering old information
  - i_t: Input gate vector: weight of acquiring new information
  - o_t: Output gate vector: Output candidate

Fig. 3: A Long Short-Term Memory Cell.

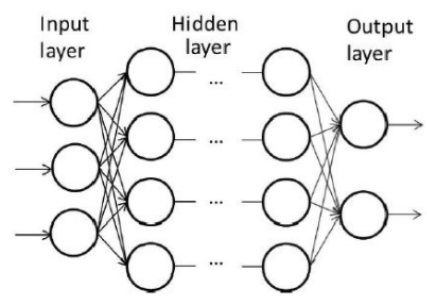


Fig. 4: Multilayer perceptron architecture.

IV. Experimental data

The experiments are done using the Spoken Arabic Digit dataset grouping the Arabic digit from 0 to 9 as illustrated in figure 5 [4]. This dataset contains time series of MFCCs corresponding to spoken Arabic digits collected by the laboratory of automatic and signals, University of Badji-Mokhtar - Annaba, Algeria. A number of 88 individual (44 males and 44 females) Arabic native speakers were asked to utter all digits ten times. Accordingly, the database consists of 8800 tokens (10 digits x 10 repetitions x 88 speakers) [8]. The whole dataset is divided into two parts: a training set

Digit	Arabic Writing
0	صفر
1	واحد
2	إثنان
3	ثلاثة
4	أربعة
5	خمسة
6	ستة
7	سبعة
8	ثمانية
9	تسعة

Fig. 5: Arabic Digit and their writing.

with 75% of the samples (6600 tokens) and the test set with the remaining 25% samples (2200 tokens).

The Mel Frequency Cepstral Coefficients were computed with the following parameters illustrate in Table I.

TABLE I: MFCC Parameters used in the experiments.

Parameters	Values
Sampling rate	11025 Hz, 16 bits
Filter pre-emphasized	1-0.97*Z^{-1}
Applied window	Hamming

V. Performance criteria

Different performance measures can be used to evaluate the performance of the ASR systems [24]. In our study, we have adopted the following standard classification criteria: recall, precision, f-measure (F1), % of error and % of success. They are defined below:

$$precision = \frac{\text{number of correct predictions}}{\text{number of predictions}}$$

$$recall = \frac{\text{number of correct predictions}}{\text{number of samples}}$$

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

VI. Experimental setup of the end-to-end encoder-classifier

The proposed end-to-end neural network that takes as input the sequences of MFCC features and as output the class of the spoken digit. First, the network will encode the sequence of MFCC coefficients as a fixed size vector. This fixed sized vector will feed an MLP network to classify the MFCC coefficients. To do so, we, first, encode the data as a matrix of (6600,93,13) where 6600 is the size of samples in our training data, 93 is the size of the longest sequence of MFCC coefficients (corresponding to the long time of recording) and 13 is the number of MFCC coefficients used in this study. When the size of the sequence is smaller than 93, the sequence is padded by a zeroed vector of size 13 until reaching the maximum size of 93.

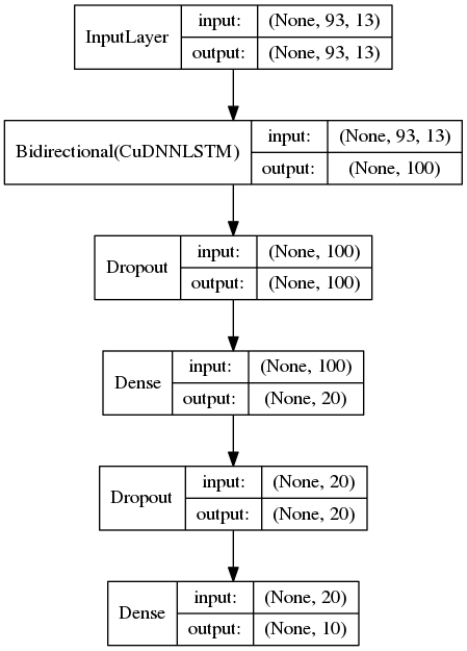


Fig. 6: Block scheme of the proposed end-to-end model.

Next, the data are processed by a bidirectional LSTM layer to encode the sequence as a fixed size vector, we choose a bidirectional model since it has been shown that bidirectional models are in general more efficient than single direction models as it is proved by the results reported in table III [18]. The vector is then directed to a multilayer perceptron structure with one hidden layer. The diverse parameters have been fixed intuitively as follow:

- the output of the LSTM layer has been fixed to 50, the input sequence has been processed by an LSTM layer in a forward / backward direction by the other LSTM layer. The encoded fixed size vector is the concatenation of the 2 LSTM output vectors.
- hidden layer size is set to 50 and non-linear activation function is set to Rectified Linear Unit (ReLU).
- Output layer is set to 10, corresponding to the 10 classes of digits, with a standard softmax activation function using cross entropy loss.

In order to regularize the network, two Dropout layers are inserted respectively after the LSTM output and the MLP hidden layer with a dropout probability of 0.2 and 0.5 respectively [25].

This neural network is trained for 50 epochs with a batch size of 16, the final model chosen is the one that provides the best f-mesure performance on the training set. The figure 6 summarizes the topology of this neural network and the table IIa gives the performance averaged over 10 experiments of this model on the test set in terms of precision, recall and f-mesure for each class. The performance (98.77% of global f-measure) is actually close to perfect on this dataset and outperforms

TABLE II: Results and comparison of the end-to-end neural approach with some previous published approaches on the same dataset.

(a) Results with the end-to-end approach.

dig.	precision	recall	F1	%error
0	95.98	98.73	97.33	4.14
1	98.92	99.86	99.39	1.09
2	99.63	98.91	99.27	1.09
3	98.67	97.45	98.06	2.55
4	99.64	99.32	99.48	0.68
5	99.32	99.91	99.61	0.68
6	99.81	96.36	98.06	3.64
7	98.55	98.82	98.68	1.45
8	98.32	98.41	98.36	1.68
9	98.96	99.91	99.43	1.05
All	98.77	98.77	98.77	1.23

(b) Comparison with the approaches by [7] and [8] in terms of % success.

dig.	[7]	[8]	our BiLSTM
0	91.00	85.55	**95.86**
1	**99.00**	98.36	98.91
2	91.50	92.91	**98.91**
3	88.00	94.09	**97.45**
4	81.50	89.91	**99.32**
5	94.50	94.00	**99.32**
6	84.50	93.82	**96.36**
7	89.50	90.18	**98.55**
8	92.50	99.00	**98.32**
9	91.00	93.36	**98.95**
All	90.35	93.12	**98.77**

TABLE III: Results of different end-to-end approaches.

type of encoder	#params in the network	F1	%error
Bidirectionnel LSTM 2*50	31.560	98.77	1.23
Bidirectionnel GRU 2*50	25.060	98.63	1.37
Forward GRU 100	40.060	97.26	2.74
Backward GRU 100	40.060	98.85	1.15
Forward LSTM 100	51.560	97.41	2.59
Backward LSTM 100	51.560	98.33	1.67

largely the previous published results reported on the same dataset [7], [8] (see table IIb).

To confirm the choice of the bidirectional recurrent architecture as encoder, we evaluate different encoders using both combination of GRU and LSTM layers with forward, backward and bidirectional encoding strategies. We fix the size of the encoding vector at 100. We evaluate the performance of the network using forward, backward and bidirectional LSTM and GRU encoders in terms of f-measure and % error, The proposed strategies are listed below and the results are reported in the table III:

- Bidirectional LSTM of size 50;
- Bidirectional GRU of size 50;
- Forward GRU of size 100;
- Backward GRU of size 100;
- Forward LSTM of size 100;
- Backward LSTM of size 100;

All encoder variants exhibit good results and outperform those by [7] and [8] by at least 5% of accuracy. These experiments tend to confirm that bidirectional architectures are more efficient than single direction ones while having less parameters. It should be noted that, the processing sequences in backward direction looks also more efficient than those by forward direction.

VII. Conclusion

In this paper, an end-to-end approach based on recurrent neural networks to process sequences of variable lengths of MFCC features was presented. The extracted MFCC features sequences are, first, encoded as a fixed size vector by a recurrent LSTM/GRU neural network, next, the obtained vector was introduced to a standard multilayer perceptron to classify the spoken word. Based on the results and discussions presented in this paper, obtained on spoken Arabic digits, the obtained results confirm the effectiveness of the proposed model. In all the experiments carried out, the proposed system presents an improved performance with respect to some existing models in the literature and obtains promising results. The obtained results show that MFCC features (introduced to a classification system) are efficient enough to characterize the speech signal for this kind of tasks. The Challenge for future works, is to assess this kind of systems with noisy (more realistic) speech signals.

Acknowledgments

The authors express their gratitude to the dedicated personnel who made the Arabic digits dataset used in this study freely available. They are also grateful for the support of NVIDIA Corporation with the donation of the GTX Titan X GPU used in this research work.

References

[1] K. Davis, R. Biddulph, and S. Balashek, "Automatic recognition of spoken digits," *The Journal of the Acoustical Society of America*, vol. 24, no. 6, pp. 637–642, 1952.

[2] N. Desai, K. Dhameliya, and V. Desai, "Feature extraction and classification techniques for speech recognition: A review," *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 12, pp. 367–371, 2013.

[3] B.-H. Juang and L. Rabiner, *Fundamentals of speech recognition*. Prentice-Hall, 1993.

[4] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[5] D. Gupta, P. Bansal, and K. Choudhary, "The state of the art of feature extraction techniques in speech recognition," in *Speech and Language Processing for Human-Machine Communications*. Springer, 2018, pp. 195–207.

[6] K. Saeed and M. K. Nammous, "A speech-and-speaker identification system: feature extraction, description, and classification of speech-signal image," *IEEE transactions on Industrial electronics*, vol. 54, no. 2, pp. 887–897, 2007.

[7] N. Hammami and M. Sellam, "Tree distribution classifier for automatic spoken arabic digit recognition," in *Internet Technology and Secured Transactions, 2009. ICITST 2009. International Conference for*. IEEE, 2009, pp. 1–4.

[8] N. Hammami and M. Bedda, "Improved tree model for arabic speech recognition," in *2010 3rd International Conference on Computer Science and Information Technology*, vol. 5, July 2010, pp. 521–526.

[9] K. Daqrouq, M. Alfaouri, A. Alkhateeb, E. Khalaf, and A. Morfeq, "Wavelet lpc with neural network for spoken arabic digits recognition system," *British Journal of Applied Science & Technology*, vol. 4, no. 8, p. 1238, 2014.

[10] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, 2013, pp. 6645–6649.

[11] D. AbuZeina and M. Elshafei, "Arabic speech recognition systems," in *Cross-Word Modeling for Arabic Speech Recognition*. Springer, 2012, pp. 17–23.

[12] G. Mamta, A. A. Shatru, and G. Savita, "Noise robust speech recognition system using mel cepstral and genetic algorithm," in *Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on*. IEEE, 2016, pp. 3151–3155.

[13] D. A. Reynolds, "An overview of automatic speaker recognition technology," in *Acoustics, speech, and signal processing (ICASSP), 2002 IEEE international conference on*, vol. 4. IEEE, 2002, pp. IV–4072.

[14] W. Han, C.-F. Chan, C.-S. Choy, and K.-P. Pun, "An efficient mfcc extraction method in speech recognition," in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*. IEEE, 2006, pp. 4–pp.

[15] Z. Ali, A. W. Abbas, T. Thasleema, B. Uddin, T. Raaz, and S. A. R. Abid, "Database development and automatic speech recognition of isolated pashto spoken digits using mfcc and k-nn," *International Journal of Speech Technology*, vol. 18, no. 2, pp. 271–275, 2015.

[16] M. Anusuya and S. Katti, "Front end analysis of speech recognition: a review," *International Journal of Speech Technology*, vol. 14, no. 2, pp. 99–145, 2011.

[17] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," in *International Conference on Machine Learning*, 2015, pp. 2067–2075.

[18] V. Vukotic, C. Raymond, and G. Gravier, "A step beyond local observations with a dialog aware bidirectional GRU network for Spoken Language Understanding," in *Interspeech*, San Francisco, United States, September 2016. [Online]. Available: https://hal.inria.fr/hal-01351733

[19] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[20] Y. Gao and D. Glowacka, "Deep gate recurrent neural network," in *Asian Conference on Machine Learning*, 2016, pp. 350–365.

[21] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *arXiv preprint arXiv:1508.01991*, 2015.

[22] S. K. Pal and S. Mitra, "Multilayer perceptron, fuzzy sets, and classification," *IEEE Transactions on neural networks*, vol. 3, no. 5, pp. 683–697, 1992.

[23] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[24] H. Jiang, "Confidence measures for speech recognition: A survey," *Speech communication*, vol. 45, no. 4, pp. 455–470, 2005.

[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014. [Online]. Available: http://dl.acm.org/citation.cfm?id=2627435.2670313